In the Claims

1. [Currently Amended] A system for processing boundary information of a graphical object, comprising:

code for receiving a graphical image that comprises said graphical object , wherein said graphical object is defined by at least said boundary information;

code for detecting a plurality of ~~contour~~ contours between [[a]] respective ~~pair~~ pairs of points of said graphical image, wherein individual ones of the contours are detected responsive to respective user input of a user; and

code for determining a plurality of vertices from said boundary information, wherein respective contours, which are between adjacent vertices of said plurality of vertices detected by said code for detecting, approximate respective edges of said boundary information within a distortion criterion.

2. [Currently Amended] The system of claim 1 further comprising:

code for creating an approximated boundary utilizing at least said graphical image, said plurality of vertices, and said code for detecting.

3. [Currently Amended] The system of claim 1 wherein ~~said~~ the code for detecting comprises a predetermined function [[is]] operable to calculate gradients associated with said graphical image.

4. [Original] The system of claim 3 wherein said code for detecting is operable to determine a shortest path between said pair of points, wherein said shortest path is weighted by said calculated gradients.

5. [Original] The system of claim 4 wherein said code for detecting limits its determination of the shortest path to a rectangular area defined in part by a width parameter.

6. [Original] The system of claim 3 wherein said calculated gradients are calculated over respective spatial areas of said graphical image limited by a scale parameter.

7. [Original] The system of claim 1 wherein said code for detecting implements a Rubberband function in executable instructions.

8. [Original] The system of claim 1 wherein said code for determining only analyzes points of said boundary information that are associated with respective edges that are less than a heuristic value.

9. [Original] The system of claim 1 wherein said code for determining only analyzes vertex pairs associated with edges of an edge set that is a weighted acyclic graph.

10. [Original] The system of claim 1 wherein said code for determining a plurality of vertices only analyzes vertices from a searchable set of vertices.

11. [Original] The system of claim 1 wherein said searchable set of vertices only includes: (a) vertices associated with curvature greater than a first heuristic value and (b) vertices recursively grown by maximizing distances between adjacent vertices subject to the following constraints: (i) said maximizing distances are less than a second heuristic value and (ii) each contours between adjacent vertices detected by said code for detecting approximate respective edges of said boundary information within a distortion criterion.

12. [Currently Amended] A method for processing boundary information of a graphical object, comprising:

receiving a graphical image that comprises said graph graphical object, wherein said graphical object is defined by at least said boundary information;

determining a plurality of vertices from said boundary information, wherein adjacent vertices of said plurality of vertices are associated with respective contours that approximate respective edges of said boundary information within a distortion criterion, wherein said respective contours are detected by analysis of said graphical image by a predetermined function and responsive to different user input for respective individual ones of the contours;

and

encoding at least said plurality of vertices in a data structure to represent said boundary information.

13. [Original] The method of claim 12 wherein said predetermined function is operable to determine a shortest path between adjacent vertices, wherein said shortest path is weighted by gradients calculated from said graphical image.

14. [Original] The method of claim 13 wherein said predetermined function is operable to determine said shortest path from only a spatial area defined by at least a width parameter.

15. [Original] The method of claim 13 wherein said predetermined function is operable to calculate said gradients utilizing a pixel neighborhood defined by a scale parameter.

16. [Original] The method of claim 12 wherein said determining comprises identifying a point of said boundary information that is associated with a greatest amount of curvature.

17. [Original] The method of claim 12 wherein said determining only analyzes vertex pairs associated with edges that are shorter than a heuristic value.

18. [Original] The method of claim 12 wherein said determining only selects vertices from a searchable set of vertices.

19. [Original] The method of claim 18 wherein said searchable set of vertices only includes: (a) vertices associated with curvature greater than a first heuristic value and (b) vertices recursively grown by maximizing distances between adjacent vertices subject to the following constraints: (i) said maximizing distances are less than a second heuristic value and (ii) respective

contours between adjacent vertices approximate respective edges of said boundary information within a distortion criterion.

20. [Original] A method for processing boundary information associated with an object in a graphical image, said method comprising:

identifying two vertices in said graphical image;

detecting a plurality of contours between said two vertices by determining a respective shortest path between said two vertices, said respective shortest path being weighted by gradient calculations of said graphical image over regions defined at least by a scale parameter, and each contour of said plurality of contours being associated with a respective scale parameter of a plurality of scale parameters; and

selecting an optimal scale parameter from said plurality of scale parameters by determining a scale parameter from said plurality of scale parameters that minimizes variance between regions defined by its respective contours.

21. [Original] The method of claim 20 wherein said method further comprising:

encoding a boundary object utilizing said two vertices and said optimal scale parameter.

22. [Original] The method of claim 20 wherein said detecting further comprising:

incrementally detecting a contour of said plurality of contours by utilizing a threshold value, wherein said shortest path is determined by a graph searching process that limits searching of paths to distances less than said threshold value.

23. [Currently Amended] The method of claim 20 wherein said detecting a plurality of contours is operable to only select contours within a rectangular area defined by a width ~~parameters~~ parameter and said two vertices.

24. [Currently Amended] The method of claim 23 wherein said width ~~parameters~~ parameter and said two vertices are selected by a user interface.

25. [New] The system of claim 1 wherein the user input is different for individual ones of the contours.

26. [New] The system of claim 1 wherein the user input selects an area of the graphical image wherein searching for the contours is performed.

27. [New] The system of claim 26 wherein the graphical image has an associated area, and the selected area comprises an area less than an entirety of the area of the graphical image.

al amt

28. [New] The system of claim 26 wherein the user input selects, for individual ones of the contours, at least one of the respective vertices and a width of the area.

29. [New] The method of claim 12 wherein the user input is different for individual ones of the contours.

30. [New] The method of claim 12 wherein the predetermined function comprises a Rubberband function.

31. [New] The method of claim 20 wherein the detected contours approximate respective edges of the boundary information.

32. [New] The method of claim 31 wherein the edges of the boundary information exist before the detecting.

33. [New] A computer comprising:

a display configured to depict a graphical image comprising a graphical object;

a user interface configured to receive user input; and

processing circuitry coupled with the display and configured to determine a plurality of vertices using boundary information of the graphical object, and to detect a plurality of contours between respective pairs of the vertices responsive to respective user input received via the user interface, wherein the contours approximate respective edges of the boundary information of the graphical object within a distortion criterion.

34. [New] The computer of claim 33 wherein the user input is different for individual ones of the contours.